

①9 RÉPUBLIQUE FRANÇAISE  
INSTITUT NATIONAL  
DE LA PROPRIÉTÉ INDUSTRIELLE  
PARIS

①1 N° de publication :

2 783 065

(à n'utiliser que pour les  
commandes de reproduction)

②1 N° d'enregistrement national :

98 11145

⑤1 Int Cl<sup>7</sup> : G 06 F 9/30, H 03 M 7/00 // G 06 K 19/07

⑫

## DEMANDE DE BREVET D'INVENTION

A1

②2 Date de dépôt : 07.09.98.

③0 Priorité :

④3 Date de mise à la disposition du public de la  
demande : 10.03.00 Bulletin 00/10.

⑤6 Liste des documents cités dans le rapport de  
recherche préliminaire : *Se reporter à la fin du  
présent fascicule*

⑥0 Références à d'autres documents nationaux  
apparentés :

⑦1 Demandeur(s) : DE LA RUE CARTES ET SYSTEMES  
Société par actions simplifiée — FR..

⑦2 Inventeur(s) : CHAMLEY OLIVIER.

⑦3 Titulaire(s) :

⑦4 Mandataire(s) : CABINET BONNET THIRION.

⑤4 PROCÉDE DE COMPRESSION DE CODE INTERPRETABLE.

⑤7 Le procédé de réduction de taille mémoire occupé par  
le code d'une application interprétée, mémorisé au sein  
d'une carte à puce dotée d'un interpréteur, comporte les  
étapes suivantes:

- lecture du code interprété de l'application, et recon-  
naissance, dans le code interprété, des séquences dupli-  
quées à l'identique au moins deux fois,

- pour au moins un groupe de séquences dupliquées de  
code interprété, définition d'une portion de code interprété  
identique à ces séquences dupliquées, appelée "séquence  
de compression", et mémorisation de cette séquence de  
compression

- remplacement d'au moins une séquence dupliquée  
d'un groupe de séquences dupliquées dans le code inter-  
prété original par une séquence d'appel de la séquence de  
compression correspondante.

code après  
remplacement

appel SDC N

appel SDC N

FR 2 783 065 - A1



# "Procédé de compression de code interprétable"

Le procédé objet de la présente invention est du domaine des procédés de compression de code informatique. Elle concerne plus particulièrement les applications exécutées dans des cartes à puce et les dispositifs disposant d'un interpréteur de code, tel que machines virtuelles "Java".

5 La technologie actuelle des microcircuits pour cartes à puce ne permet encore que de faibles capacités pour les différentes mémoires du microcircuit, ce qui incite à rechercher la diminution de la taille des applications chargées dans les mémoires de ces cartes à puce. Ces mémoires sont essentiellement de type soit non volatile modifiable (par exemple de type dit "EEPROM", "flash",  
10 etc.), soit non volatile non modifiable (par exemple de type dit "ROM"), soit mémoires vives à accès direct (de type dit "RAM"). On est donc en présence d'une limitation de capacité mémoire, ce qui limite également la taille et le nombre des applications réalisables avec les cartes à puce actuelles.

La présente invention entend donc remédier à cet inconvénient en  
15 proposant un procédé de compression de code interprété mémorisé dans la carte à puce, de manière à réduire sensiblement la taille mémoire occupée par une application donnée.

Selon un second objectif de l'invention, le procédé est simple de mise en œuvre et rapide d'exécution. Il s'applique à des dispositifs de carte à puce  
20 existants sans modification.

Le procédé selon l'invention est donc un procédé de réduction de taille mémoire occupé par le code d'une application, mémorisé dans un dispositif adapté au sein d'une carte dite "à puce", ladite carte étant dotée d'un interpréteur dit "machine virtuelle", l'application étant interprétée par la machine  
25 virtuelle, caractérisé en ce qu'il comporte les étapes suivantes :

- lecture du code interprété de l'application, et reconnaissance, dans le code interprété, des séquences dupliquées à l'identique au moins deux fois,
- pour au moins un groupe de séquences dupliquées de code interprété, définition d'une portion de code interprété identique à ces séquences

dupliquées, appelée "séquence de compression", et mémorisation de cette séquence de compression (SDC),

- remplacement d'au moins une séquence dupliquée d'un groupe de séquences dupliquées dans le code interprété original par une séquence d'appel de la séquence de compression (SDC) correspondante.

On comprend que l'utilisation de compression de code dans le domaine des applications de carte à puce répond au problème de limitation de capacité mémoire de ces cartes.

Par ailleurs, cette disposition repose sur une analyse par séquences d'octets, au contraire des techniques classiques de compression, qui "travaillent" sur des séquences de bits, sans tenir compte de leur signification. La méthode proposée tient compte de ce que les séquences d'octets du code interprété représentent les instructions d'une machine virtuelle.

On note également que ce procédé permet effectivement de réduire la taille occupée par une application écrite en code interprété par une machine virtuelle implémentée dans une carte à puce, sans augmentation significative du jeu d'instructions de la machine virtuelle.

Selon une mise en œuvre préférée, la séquence d'appel de la séquence de compression est un octet unique de code interprété réalisant l'appel de la séquence de compression.

Cette disposition est favorable à une bonne réduction de taille du code interprété.

Selon une mise en œuvre préférée, la définition de séquence de compression est réalisée pour des groupes de séquences dupliquées de code interprété, vérifiant certains critères. Préférentiellement, les critères appliqués aux séquences dupliquées comportent en particulier les points suivants :

- ne pas contenir d'instruction de saut aboutissant en dehors de la séquence de compression (SDC),
- ne pas accepter qu'une instruction de saut aboutisse à l'intérieur d'une séquence de compression (SDC),

- ne pas contenir de code résultant de l'exécution de code interprété en sous-programme.

Ces dispositions permettent de réaliser une implémentation du procédé restreinte aux cas de compression simples.

5        La description et les dessins qui suivent permettront de mieux comprendre les buts et avantages de l'invention. Il est clair que cette description est donnée à titre d'exemple, et n'a pas de caractère limitatif. Dans les dessins :

- les figures 1a et 1b représentent schématiquement un code interprété avant (figure 1a) et après (figure 1b) mise en œuvre du procédé ;
- 10        - la figure 2 représente la table des pointeurs et la zone des séquences de compression dans la mémoire du microcircuit.

Dans le cadre de la description donnée ici à titre non limitatif, l'assemblage décrit est celui d'une carte plastique de type " carte à puce " à machine virtuelle, de type connu de l'homme de l'art, dans lesquelles des applications interprétées sont chargées dynamiquement dans la mémoire non volatile modifiable du microcircuit (par exemple de type dit "EEPROM" ou "Flash"). Elle s'applique tout aussi bien aux applications interprétées inscrites en fabrication dans la mémoire non volatile non modifiable du microcircuit (de type ROM).

20        Le procédé réalise la compression du code d'une application interprété destiné à être mémorisé dans une carte dite à puce dotée d'un interpréteur ("machine virtuelle"). Le procédé est alors mis en œuvre de la manière suivante (par exemple par un système informatique dans lequel le code interprété est mémorisé selon une méthode et avec des dispositifs connus en soi):

25        1/ Le code interprété de l'application est tout d'abord parcouru, et on procède à une reconnaissance, dans le code interprété, des séquences (de longueur minimale 2 octets), dupliquées à l'identique au moins deux fois. Cette reconnaissance est réalisée selon des algorithmes connus de l'homme de métier et hors du champ de la présente invention.

30        2/ Pour chaque groupe de séquences dupliquées de code interprété, il est défini une portion de code interprété identique à ces séquences dupliquées,

appelée "séquence de compression" (SDC), et on enregistre en mémoire au sein du microcircuit de la carte à puce cette séquence de compression,

5 3/ En revenant alors au code interprété original, on procède au remplacement de chaque séquence dupliquée dans le code interprété original par une séquence d'appel de la séquence de compression (SDC) correspondante.

Tel que représenté sur la figure 1, une implémentation particulière du procédé selon l'invention consiste à désigner une séquence de compression (SDC) par la valeur N de l'index correspondant de la table des pointeurs (figure  
10 2) sur la zone des SDC, et à ajouter les instructions suivantes au jeu d'instructions de la machine virtuelle :

- "appel court SDC n° N", avec N compris entre 0 et 15, N faisant partie de l'octet d'instruction, constitué de 4 bits de code opératoire et de 4 bits représentant N,
- 15 - "appel long SDC n° N", avec N supérieur à 15, et nécessitant un octet d'opérande à la suite de l'octet d'instruction pour contenir N,
- "retour de SDC".

Cette troisième instruction, de longueur un octet, n'est d'ailleurs pas nécessaire dans le cas où les SDC sont stockées de manière contiguë dans  
20 l'espace mémoire réservé aux SDC, ce qui est le cas dans l'implémentation décrite, car la différence entre les valeurs des pointeurs sur deux séquences de compression consécutives  $SDCN+1$  et  $SDCN$  fournit alors la longueur de la séquence  $SDCN$ , ce qui permet de déterminer le retour de  $SDCN$  sans l'aide d'instruction "retour de SDC".

25 Dans cette implémentation, en supposant qu'un code interprété fait appel A fois à une séquence de compression  $SDCN$  contenant L octets, la réduction de taille de code interprété est de G octets par rapport au code sans création de séquence de compression, avec :

$$G = A \times L \text{ (taille mémoire dans le code interprété original)} - A \text{ (octets d'appels à la SDCN insérés dans le code transformé)} - L \text{ (longueur de la SDCN)}$$
  
30

dans l'espace mémoire) - 1 (octet de retour de la SDCN) - 2 (octets de la table des pointeurs sur la SDCN),

soit  $G = (A-1) \times (L-1) - 4$

5 Par exemple, pour une séquence de compression comportant 4 octets et appelée 5 fois, le gain est de 8 octets. Il est évident que si le gain est inférieur à zéro, le code initial n'est pas transformé.

Pour obtenir une implémentation simple, des restrictions sont imposées aux séquences de compression (SDC), et en particulier les suivantes :

- ne pas contenir d'instruction de saut aboutissant en dehors de la SDC,
- 10 - ne pas accepter qu'une instruction de saut aboutisse à l'intérieur d'une SDC (c'est-à-dire refuser de transformer en SDC des séquences au milieu desquelles aboutit une instruction de saut),
- ne pas contenir de code résultant en l'exécution de code interprété en sous-programme (la récursivité des séquences de compression, c'est-à-dire l'appel d'une SDC dans une SDC, n'est pas supportée). L'appel de code natif
- 15 reste néanmoins autorisé,
- ne pas se terminer entre un code opératoire et son (ses) opérande(s).

20 Une fois le code transformé selon le procédé décrit, ce code est mémorisé dans le microcircuit de la carte à puce à machine virtuelle de façon classique.

On comprend que ce procédé présente l'avantage de ne pas augmenter significativement le jeu d'instructions de la machine virtuelle, à chaque création d'une séquence de compression, puisque le procédé n'ajoute que deux ou trois instructions nouvelles à la machine virtuelle.

25 Il est à noter que les SDC se distinguent à la fois des sous-programmes et des macro-instructions. En effet, contrairement aux sous-programmes, l'appel à la SDC et le retour de la SDC n'utilisent pas la pile du microcircuit, et cette pile peut être dans des états différents lors de l'appel à la SDC et lors du retour de la SDC, (ce qui ne serait pas le cas avec un sous-programme classique). Vis-à-vis

30 d'une macro-instruction, lors du traitement par un macro-processeur associé, la

macro-instruction est remplacée par la séquence de code qu'elle représente, chaque fois qu'elle apparaît dans le code source initial.

5 La portée de la présente invention ne se limite pas aux détails des formes de réalisation ci-dessus considérés à titre d'exemple, mais s'étend au contraire aux modifications à la portée de l'homme de l'art.

## REVENDEICATIONS

1. Procédé de réduction de taille mémoire occupé par le code d'une application, mémorisé dans un dispositif adapté au sein d'une carte dite "à puce", ladite carte étant dotée d'un interpréteur dit "machine virtuelle", l'application étant interprétée par la machine virtuelle, caractérisé en ce qu'il  
5 comporte les étapes suivantes :

- lecture du code interprété de l'application, et reconnaissance, dans le code interprété, des séquences dupliquées à l'identique au moins deux fois,
- pour au moins un groupe de séquences dupliquées de code interprété, définition d'une portion de code interprété identique à ces séquences  
10 dupliquées, appelée "séquence de compression", et mémorisation de cette séquence de compression (SDC),

- remplacement d'au moins une séquence dupliquée d'un groupe de séquences dupliquées dans le code interprété original par une séquence d'appel de la séquence de compression (SDC) correspondante.

15 2. Procédé selon la revendication 1, caractérisé en ce que la séquence d'appel de la séquence de compression est un octet unique de code interprété réalisant l'appel de la séquence de compression.

20 3. Procédé selon la revendication 2, caractérisé en ce que la définition de séquence de compression est réalisée pour des groupes de séquences dupliquées de code interprété, vérifiant certains critères.

4. Procédé selon la revendication 3, caractérisé en ce que les critères appliqués aux séquences dupliquées comportent en particulier les points suivants :

- ne pas contenir d'instruction de saut aboutissant en dehors de la  
25 séquence de compression (SDC),

- ne pas accepter qu'une instruction de saut aboutisse à l'intérieur d'une séquence de compression (SDC),



- ne pas contenir de code résultant en l'exécution de code interprété en sous-programme,
- ne pas se terminer entre un code opératoire et son (ses) opérande(s).

code avant  
remplacement

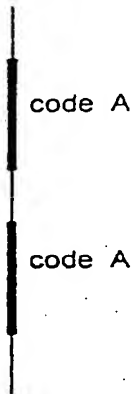


FIG 1a

code après  
remplacement

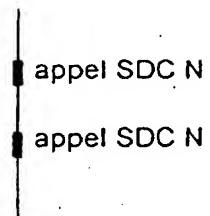


FIG 1b

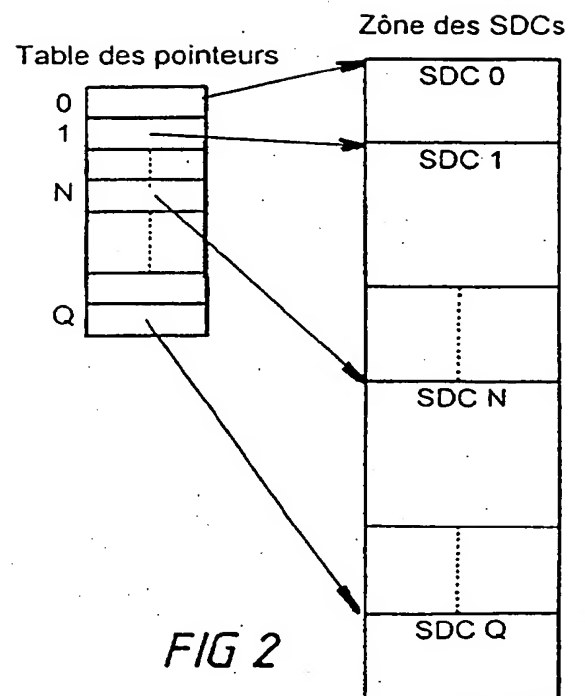


FIG 2

INSTITUT NATIONAL  
de la  
PROPRIÉTÉ INDUSTRIELLE

RAPPORT DE RECHERCHE  
PRELIMINAIRE

établi sur la base des dernières revendications  
déposées avant le commencement de la recherche

N° d'enregistrement  
national

FA 565364  
FR 9811145

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
X	VAHID F: "Procedure exlining: a transformation for improved system and behavioral synthesis" PROCEEDINGS OF THE EIGHTH INTERNATIONAL SYMPOSIUM ON SYSTEM SYNTHESIS (IEEE CAT. NO.95TH8050), PROCEEDINGS OF THE EIGHTH INTERNATIONAL SYMPOSIUM ON SYSTEM SYNTHESIS, CANNES, FRANCE, 13-15 SEPT. 1995, pages 84-89, XP002098348 ISBN 0-8186-7076-2, 1995, Los Alamitos, CA, USA, IEEE Comput. Soc. Press, USA	1
A	* page 508, colonne de droite, ligne 14 - page 509, colonne de gauche, ligne 7 * * page 509, colonne de gauche, ligne 47 - ligne 55 * * alinéa 3.1 * * alinéa 4 *	2,3
A	GUTHERY: "JAVA CARD: Internet Computing on a Smart Card" IEEE INTERNET COMPUTING, février 1997, page 57 57 XP002077647 * le document en entier *	1
A	US 5 481 708 A (KUKOL PETER ET AL) 2 janvier 1996 * colonne 7, ligne 43 - ligne 50 * * colonne 11, ligne 12 - ligne 20 * --- -/--	1,2
Date d'achèvement de la recherche		Examineur
30 mars 1999		Bijn, K
<p>CATEGORIE DES DOCUMENTS CITES</p> <p>X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intercalaire</p> <p>T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons &amp; : membre de la même famille, document correspondant</p>		

2

EPO FORM 1503 82.82 (P04C13)

INSTITUT NATIONAL  
de la  
PROPRIETE INDUSTRIELLE

RAPPORT DE RECHERCHE  
PRELIMINAIRE  
établi sur la base des dernières revendications  
déposées avant le commencement de la recherche

N° d'enregistrement  
national

FA 565364  
FR 9811145

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
A	<p>BAKER B S: "On finding duplication and near-duplication in large software systems" PROCEEDINGS. SECOND WORKING CONFERENCE ON REVERSE ENGINEERING (CAT. NO.95TB8101), PROCEEDINGS OF 2ND WORKING CONFERENCE ON REVERSE ENGINEERING, TORONTO, ONT., CANADA, 14-16 JULY 1995, pages 86-95, XP002098349 ISBN 0-8186-7111-4, 1995, Los Alamitos, CA, USA, IEEE Comput. Soc. Press, USA * page 86, colonne de droite, ligne 9 - ligne 24 *</p>	1
		DOMAINES TECHNIQUES RECHERCHES (Int.CL.6)
Date d'achèvement de la recherche		Examineur
30 mars 1999		Bijn, K
<p>CATEGORIE DES DOCUMENTS CITES</p> <p>X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intercalaire</p> <p>T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons &amp; : membre de la même famille, document correspondant</p>		

2

EPO FORM 1503 03.92 (P04C13)